E-learning Lifecycle Costs: Up-Front Decisions for Managers and ISDs

Dan Young
SCCI
Austin, TX
dyoung@teamscci.com

ABSTRACT

When training managers and instructional designers assess e-learning projects, it is well understood that the volatility of the subject matter will directly impact maintenance costs, for the simple reason that subject-matter areas with a high rate of change require more frequent content updates. There are well-known approaches using web technologies and modular design that can mitigate these costs when course content is anticipated to be volatile.

Less well understood, however, is how changes to requirements impact lifecycle costs, and how technology and course design can be used to mitigate costs when this kind of change occurs. Changes to audience, course length, level of interactivity, hosting or delivery environment, assessment methodology, delivery language, branding, and course scope are common lifecycle events in today's e-learning landscape. Typically, however, changes to requirements are seldom anticipated or designed for at the outset of an e-learning project. Instead, requirements are assumed to be static, with the result that if and when they do change, the costs can be high.

This paper will present original research demonstrating that changes to e-learning requirements occur routinely over the product lifecycle, and it will examine some of the impact associated with different kinds of changes. An analogy will be drawn between content volatility (where future maintenance costs are routinely anticipated and mitigated) and changes to requirements, where such changes should also be (but seldom are) anticipated and mitigated. Quantitative survey results indicating the frequency of different kinds of requirements changes will be shown, and mitigation strategies will be presented, including the roles of template-based design, SCORM, XML content storage, and Simplified Technical English.

ABOUT THE AUTHOR

Dan Young is a Senior Training Solutions Architect with SCCI, and has worked in DoD-related industries for eighteen years. For the last ten years, he has specialized in computer-based training, where his work involves standards research, multimedia development, process design, product management, and technical infrastructures for e-learning. He began working with the U.S. Navy's Tomahawk program in 2006, supporting SCORM compliance efforts, new technology research, and business processes for multimedia production.

E-learning Lifecycle Costs: Up-Front Decisions for Managers and ISDs

Dan Young
SCCI
Austin, TX
dyoung@teamscci.com

INTRODUCTION

Today's rapid pace of change has affected nearly every industry, and the e-learning industry is no exception. In the last few years, we have seen a number of technology shifts and evolving standards. The migration of stand-alone training to web-based training, the emergence of SCORM and learning management systems, the new affordability of Flash and other multimedia tools, Section 508 compliance requirements, and the IMS QTI standard have impacted many of our projects. Those of us who manage e-learning have become accustomed to adapting courseware to new requirements, rehosting, rebranding, and conducting upgrades, migrations and renovations. In the field of e-learning, it sometimes seems that change is the rule, and stability is the exception.

This paper imposes some numbers on this shifting landscape by presenting the results of a study that quantifies the frequency of high-impact changes to deployed courseware. In particular, it addresses changes with a global scope – requirements that impact the entire course, rather than just a local area of content. Sixteen different technical standards, design parameters, and instructional design criteria are analyzed using a data set gathered from twenty-four companies reporting on a two-year window.

The results demonstrate that high-impact change to courseware is indeed pervasive. Rather than building courseware in a fixed requirements environment, we are building courseware to meet standards and parameters that are themselves in a state of flux. This paper presents the data supporting these findings, examines some of their implications, and looks at strategies for mitigating costs in a courseware environment that is undergoing constant evolution.

THE COST OF CHANGE

Maintenance Realities: Updates and Upgrades

It is a given that the subject matter for e-learning will change over time, eventually requiring changes to the training content. Some content is more dynamic than others, but even courses that are relatively static need periodic maintenance to ensure that the training is still accurate and adequate. Organizations responsible for large courseware repositories, like the United States Navy's Navy Knowledge Online (NKO), require course owners to review their content on a regular basis to evaluate whether it needs to be updated (Navy ILE 2007). It may not be possible to predict how often the content will change, or to what extent, but we nonetheless *expect* that updates will be required in the ordinary course of e-learning maintenance. These are standard line items in the total cost of ownership.

It also sometimes happens, over the lifecycle of a course, that e-learning must adapt to meet new technical, pedagogical, or design requirements. Changes of this nature are more akin to an *upgrade* than an *update* – some aspect of the course must be reworked to fit in a new environment. The new requirement might be precipitated by pedagogical factors related to the learner or the learning objectives for this course – for instance, raising the level of interactivity, or using a simulator, rather than actual equipment, for proficiency assessments. But often, the new requirement is not directly tied to instructional design or to the learning objectives; instead, the requirement is part of a migration or upgrade agenda that involves a whole set of courses, all of which will be subject to the same requirement – for example, complying with a technical standard, implementing new branding, or adapting to a new delivery platform. These kinds of requirements originate outside of the subject matter domain, and they tend *not* to be anticipated in the ordinary cost of ownership. Rather, they reflect administrative decisions, business factors, or technical factors in the larger environment.

Regardless of whether these changes originate in policy, learner requirements, business needs, or the technical environment, changes to requirements are distinct from content updates in that their scope is global and they involve the structure (or infrastructure) of the courseware, rather than a limited area of content. Examining the nature of these requirements, outlining their cost dynamics, and quantifying their occurrence

during the e-learning maintenance lifecycle will be the focus of this paper.

Requirements: Identifying High-Impact Changes

Viewed through the lens of instructional design, requirements that impact the structure of the course should be settled at design time, prior to developing any content. This is because the labor involved in making sweeping changes grows in direct proportion to the amount of course material already completed. In fact, it can be argued that changing requirements *after* the course has been deployed, as in the maintenance scenarios that this paper will survey, is a *worst-case* scenario in terms of cost and level of effort. At maintenance time, the cost of implementing new requirements is at its *maximum* expense level.

To analyze the difference in cost dynamics between requirements change and content updates, it will be useful to briefly summarize the role of requirements in instructional systems design.

When assessing the scope of an e-learning project, instructional designers gather a variety of parameters, some dealing with instructional objectives and learner information, and others dealing with technical design specifications. This information is organized and summarized into a design document, also sometimes called a requirements document (AICC 2008), a course blueprint (Wright 2007, Greer 1992), or an instructional media design package (Navy ILE 2007). The contents of the design document vary from organization to organization, but a typical design document will include, at a minimum:

Instructional goals	Instructional strategies
Audience definition	Standards compliance
Delivery environment	User interface
Assessment strategies	Development tools
Content organization	Evaluation plan
Content sources	

(source: Cennamo & Kalk 2005)

Great care is taken in the preparation and review of this document because it serves as a guarantee to both of the parties involved. For the customer, it guarantees that the product will be engineered to mutually agreed-upon requirements. From the developer's side, it guarantees against "scope creep" – an expansion of the project beyond the agreed-upon specification, or a change to the specification that could impact the level of effort required to build the courseware.

Typically, once this document is completed, it is signed by all parties. This indicates its acceptance, but it is also a de facto acknowledgement that these requirements constitute the foundation of the entire project. Any change to these requirements could have far-reaching implications in terms of project schedule and level of effort. As such, the requirements cannot be changed without a *renegotiation* by the stakeholders involved.

This is seen most clearly in the widely used ADDIE model of instructional systems design, where course design is finalized before content development begins. But even in models that incorporate more feedback and revision in the design process (such as Dick & Carey 1996; or Kemp, Morrison & Ross 1998), the feedback is directed into the *content* rather than the *structure* of the e-learning. In these models, the learning content is dynamic, partially developed through a discovery and feedback process during development. But changes that affect requirements still require stakeholder sign-off. The reason is simple: requirements represent high-impact change.

Level of Effort

Recognizing that some maintenance requests have the kind of scope that would, in other circumstances, require renegotiating the design document helps to put the issue of e-learning maintenance in a cost -awareness context. If building a course is equivalent to building a house, there is a qualitative difference between proposals that involve simple remodeling (content) and proposals that alter the blueprint (structure). Content changes are the equivalent of the buyer saying to the builder, "Nice house, but I'd like wood flooring in the den." Requirements change is more along the lines of, "Nice house, but could we move it a few feet to the left?" The levels of effort required to support these different kinds of requests are completely out of proportion. Recognizing the difference between updates and upgrades is a critical skill in project management for e-learning.

Knowing that requirements are intrinsically high-impact, however, does not give us a simple heuristic for estimating level of effort. As part of the research for this paper, the author discussed this topic with other training managers and e-learning professionals to determine whether some changes were intrinsically higher-impact than others. When this paper was first conceived, it was thought that gathering comparative information on labor hours might show some trends in the levels of effort required by each requirement, or perhaps that requirements could be ranked in order of

impact using empirical data. After discussing this question with study participants and reviewing the data that could be gathered informally, the conclusion of this paper is that there is no simple relationship between specific requirements and level of effort. Instead, level of effort depends on a variety of factors, including course design, the tools used to build the course, and the details of the proposed change. Maintenance events that might be handled with a relatively low level of effort by one company (for example, rebranding) can be a nightmare for another.

This observation leads to a question that will be examined more closely later in this paper: if tools and course design affect the level of effort required to implement a maintenance request, are there tools and design practices that can help mitigate the cost of maintenance by creating courseware that is intrinsically more change-tolerant?

RESEARCH

Mapping the Maintenance Landscape

As we have seen, requirements occupy an almost paradoxical position in the e-learning industry. On the one hand, they are the foundation upon which e-learning is built, fixed in place before any content is developed. Yet the experience of many who manage e-learning is that requirements are constantly changing, and courses slated for updates are just as likely to include *upgrades* as part of the service request – new standards, new capabilities, changes in toolset, format, design, or environment. So what is the reality? Is it objectively the case that the business pressures governing the e-learning industry have resulted in a maintenance dynamic that resembles perpetual scope creep?

Study Parameters and Methodology

A survey was designed to provide an answer to this question. It listed sixteen different e-learning requirements, reflecting a variety of technical requirements, pedagogical requirements, standards, and other courseware maintenance activities. Respondents were asked to report how many of their e-learning courses were impacted by each requirement over the last two years, including in this figure *only* courses that had already been deployed when the requirement appeared.

The survey was deployed online, and solicitations to participate were distributed to e-learning managers, instructional designers, and e-learning course owners through several venues, including email-based communities of practice, an industry newsletter, and direct personal contacts. Twenty-eight respondents provided usable data, with roles distributed as indicated by Figure 1.

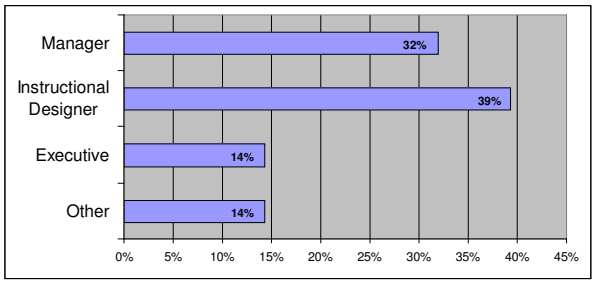

Figure 1. Breakdown of Job Roles

Some of the values in the "other" category were consultant, project lead, and "all of the above."

Approximately forty responses were rejected because of inadequate or inconsistent data. Two respondents were managers of extremely large LMS repositories. Their results, though interesting, could not be included because the quantity of their data (one repository numbering in multiple hundreds of courses, and the other in the multiple thousands) gave them an overwhelmingly disproportionate influence on the results. However, some of the findings from those repositories have been included in parts of this paper; they will be explicitly mentioned when used.

Respondents identified the markets they serve as represented on Figure 2 (note that one respondent may serve multiple markets).

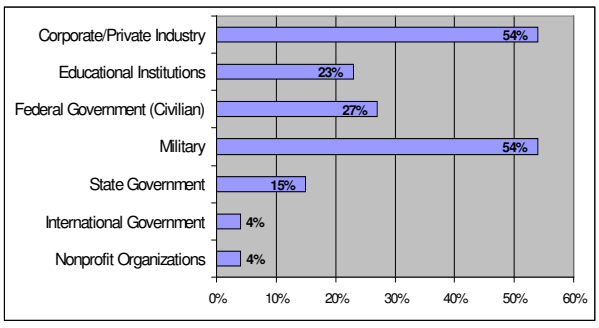

Figure 2. Breakdown of Markets Served by Respondents

Without including the two owners of very large courseware repositories, the average respondent was responsible for maintaining 20.5 courses. This figure

refers only to courses in maintenance status, excluding initial development for new courses.

Study Results

The pie chart in Figure 3 shows the percentage breakdown of the maintenance requirements included on the survey. Note that this is *not* a comprehensive breakdown of all maintenance activities undertaken by these respondents. This is only a percentage ranking of the requirements that were included in the study. (One requirement on the survey, "Becoming IMS QTI compliant" is omitted from this graph because it was not encountered as a requirement by any of the respondents; however, see the discussion of IMS QTI later in this paper for more information.)

This pie chart *does*, however, indicate the relative frequency of these requirements, as indicated by the proportion of courses impacted by each of them. As such, it serves as a quick encapsulation of where maintenance effort related to requirements is being expended.

One surprise in this data set is that changing branding ranked as the most common of the maintenance activities performed (24%). This is followed by adapting course to a new hosting or delivery environment (13%) and becoming SCORM conformant ranked third (10%). As the author of this paper is involved primarily in courseware in support of the U.S. Department of Defense, he found this ordering counter-intuitive, in that SCORM compliance is a very common requirement, and rebranding seemed to be relatively rare.

Some participants supplied their contact information and indicated that they would be willing to talk by telephone about their survey inputs, so the author asked several participants to comment on their contributions to the SCORM and rebranding data. In at least two

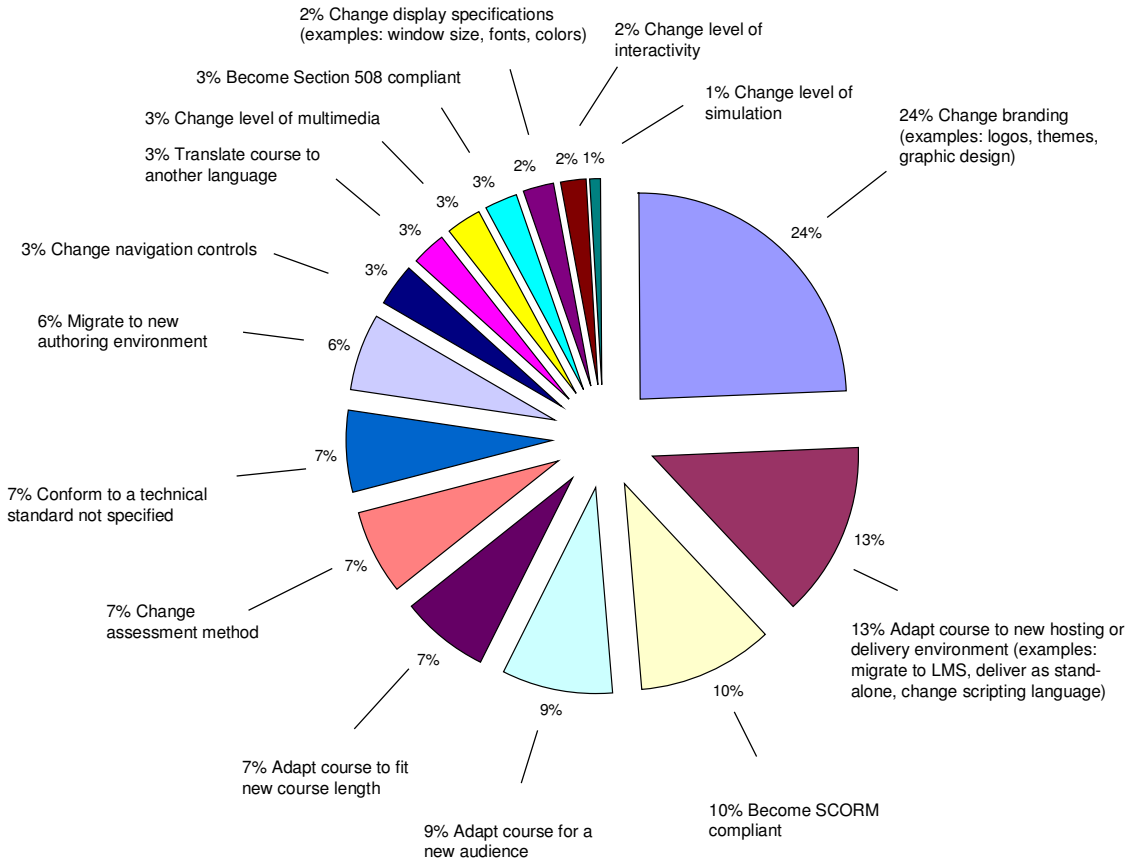

Figure 3. Breakdown of Maintenance Activities Surveyed

cases, the rebranding was undertaken in order to integrate courseware modules more seamlessly, and in at least one of those cases, the rebranded items were SCORM-conformant modules (SCOs) being reused by another course. Another respondent reported that her rebranding was the result of a corporate merger, which required extensive changes to her entire course repository in terms of branding and target audience.

Perhaps the most interesting fact about rebranding is that, though only 42% of the respondents encountered that requirement, for seven of the eleven who encountered the requirement, the change impacted 100% of their courseware. When rebranding is required, the impact affects most, if not all, courseware holdings.

With regard to the unexpectedly low incidence of SCORM conformance in this survey, several respondents reported that though they have supported SCORM conformance/conversion efforts in years prior, in the survey window it was rarer to encounter requests to convert legacy courseware. Courses that must be SCORM conformant are now typically authored from the outset using tools that support SCORM, and it is rarer now to run into conversion projects.

Detail Graphs

Figures 4, 5, and 6 contain detailed breakdowns of each requirement that was included in the survey. Three metrics, calculated by simple averages, are presented for each requirement as follows:

The *percentage of respondents* reports the percentage of the respondents who encountered the requirement at least once in the past two years.

The *percentage of courses* reports the percentage of courses that were impacted by this requirement over the last two years, out of the entire pool of all respondents' courses.

The *percentage impact for affected respondents* indicates, for all of the respondents who encountered this requirement, the percentage of their courseware that was impacted. This figure helps to identify whether some requirements tend to be applied to entire repositories (as we saw earlier, in the case of rebranding), or tend to be applied individually based on the needs of specific courses.

The results are grouped into three separate graphs – pedagogical requirements, technical requirements, and standards compliance – however, the calculations for all detail reports were performed identically and results from all three graphs may be compared against one another.

Pedagogical requirements (shown in Figure 4) relate to the target audience, teaching methodology, course scope, or the learner experience. Of all the requirements tracked, these were the most likely to originate from concerns within individual courses

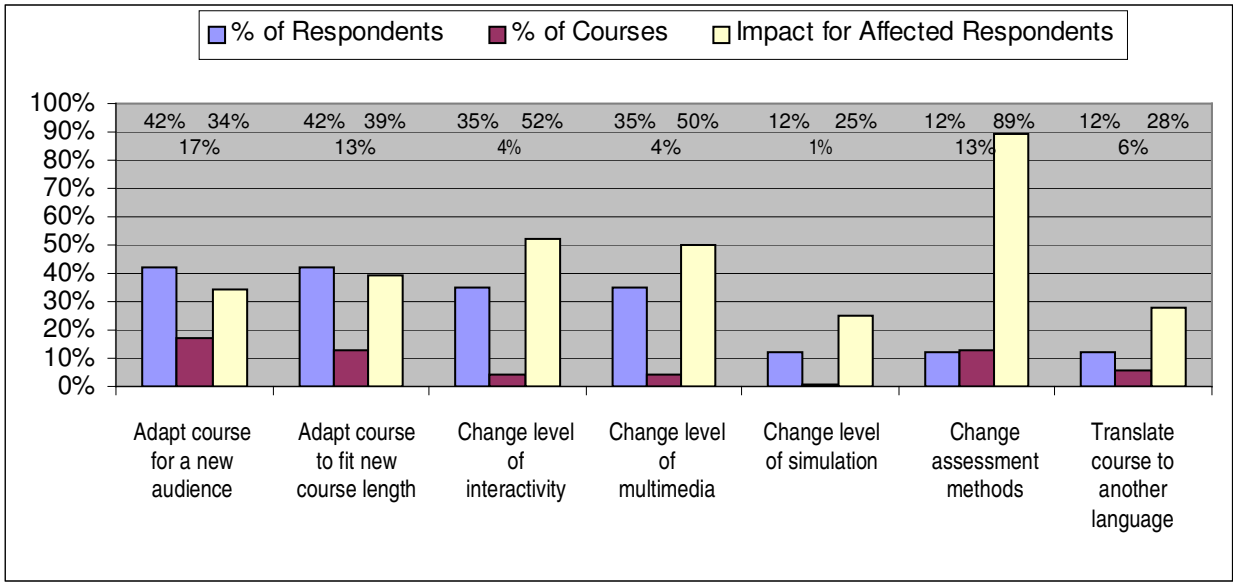

Figure 4. Pedagogical Requirements

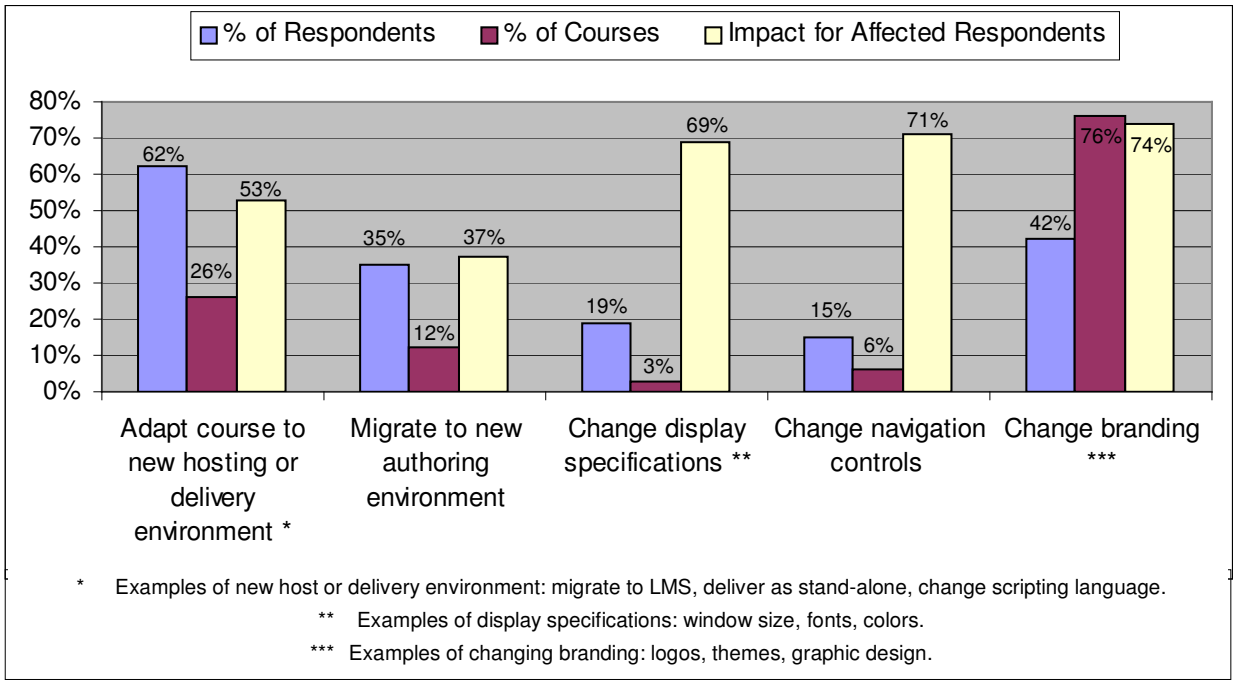

Figure 5. Technical Requirements

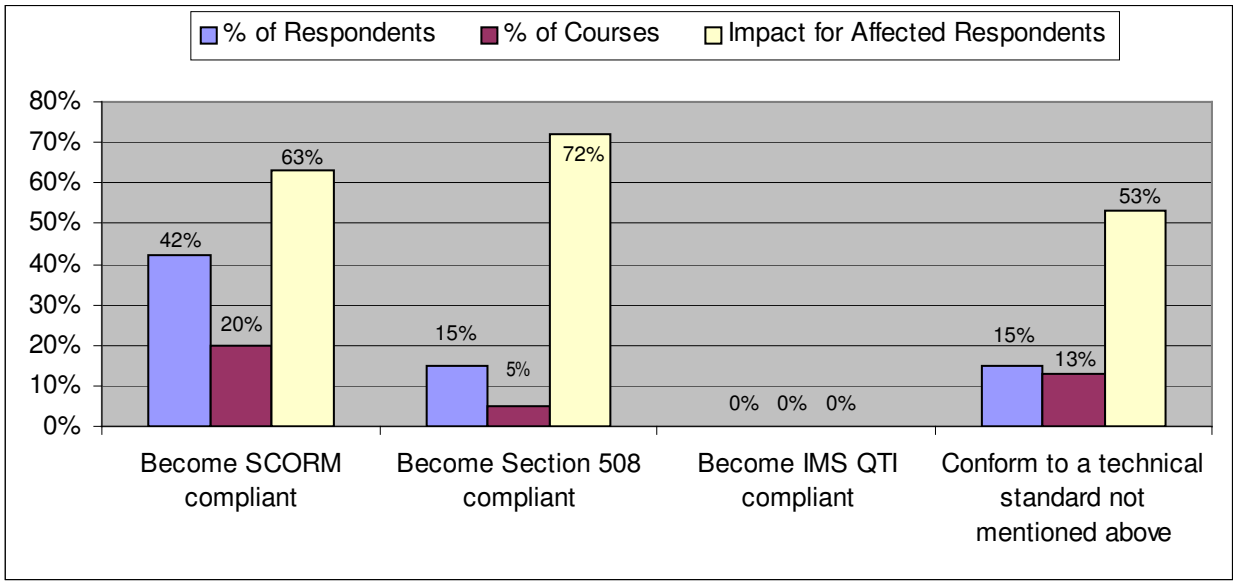

Figure 6. Standards Compliance

rather than from initiatives applied across the board. That fact is reflected in the lower *Impact for Affected Respondent* percentage. The notable exception is the requirement to change the assessment method. For the 12% of respondents who implemented this requirement, it impacted almost every course they were responsible for.

Technical requirements (shown in Figure 5) refer to the toolset, format, and infrastructure of the course. The captions for each requirement are worded exactly as they appeared on the survey, including the parenthetical examples included to help clarify the categories.

Standards compliance (shown in Figure 6) refers to SCORM, Section 508 of the Disabilities Act, and other standards. One of these standards, IMS Question Test Interoperability specification (IMS QTI), was not encountered at maintenance time by any respondents in the last two years. It is worth pointing out that this does *not* indicate that IMS QTI is not being used; it only indicates that IMS QTI is not coming up as a new requirement at maintenance time. Another data point might be relevant. A respondent who manages a government-owned courseware repository so large that it had to be excluded from the study indicated that 9% of his courses had become IMS QTI compliant in the last two years.

The Bottom Line

The data gathered here is rich in implications, but when discussing these results with other members of the e-learning community, it has been the experience of the author that each person tends to be interested in different areas of the data. We tend to be drawn to the areas where we have personal experience, and we want to compare our individual experiences with the bigger picture that emerges from aggregating the data.

In that light, rather than continuing to parse individual data items, this paper will let the graphs speak for themselves, and will move to the survey result that constitutes the bottom line of this study. This result was gathered in response to the following survey question, placed after the inventory of sixteen requirements:

Of the courses that you maintain, how many have NOT been impacted by any of the above requirements in the last two years?

This question results in the most relevant finding of this study because it asks, simply and directly, what percentage of e-learning courses, over a two-year period, were stable enough to require only content updates and routine maintenance. In other words, how stable, overall, are the tools, deployment environments, technical frameworks, and business contexts used to develop our courseware? What should our expectations be with regard to upgrading, retooling, restructuring, rebranding, redesigning, integrating, migrating, and adapting?

In this study, the average percentage for the respondents on this question was **28%**. In a two-year period, only 28% of the e-learning courses managed by these respondents were stable enough to need only content updates.

The value of this finding depends on a pair of assumptions. The first is that the two years prior to this study have not been exceptional. There is always the possibility that the current instability of the e-learning environment is the product of temporary forces.

The second assumption is that our own environment resembles the industry cross-section sampled by this study. It is possible that some corners of the industry are more dynamic in terms of standards and technology shifts, and others are more stable.

However, if we accept these two assumptions, we can invert the percentage above and apply it to the courseware we are building today. If this study is any indication, the e-learning we launch today has a 72% probability of needing high-impact maintenance sometime in the next two years.

IMPLICATIONS

The Value of Awareness

When survey participants learned about the findings of this study, some were surprised to see such a dramatic picture of change, but most, however, were not. They had, after all, supplied the data. Instead, they asked some variation of this question: *will you tell that to my boss?*

The question is serious. Unanticipated costs and labor expenditures are dogging training departments. Training managers find themselves trying to explain maintenance cost dynamics to their senior management, sometimes with only a little data to go on. In some cases, senior management may be unknowingly thinking in terms of web site maintenance models rather than a courseware model. In maintaining a web site, there is constant development activity for new content. But in that model, newer content either *replaces* older

content, or the older content is *archived* for reference. Web sites do not have "legacy" content. But as we have seen, "legacy" content is at the heart of our industry's maintenance challenge.

One e-learning manager related that her senior management noticed that her ISD team seemed to be working more slowly. Over time, the output of her instructional designers (measured in hours of instruction developed) seemed to be diminishing. It was not until the manager started drawing on the whiteboard that she realized the simple mathematical concept governing her situation: assuming fixed resources, and assuming that maintenance effort increases in direct proportion to the quantity of e-learning maintained, the more training that you develop now, the *less* you can develop in the future. Everything else being equal, when your maintenance commitments increase, your output in new courseware will drop. And, ironically, the more productive your team is, the faster and more dramatic their slowdown will be.

It does not help this situation that so many ROI studies focus only on development and deployment costs, often not taking into account routine LMS and IT operating costs, let alone courseware lifecycle concerns. Literature searches reveal that articles on e-learning maintenance are hard to find, outnumbered in orders of magnitude by literature focused on design and development. Many e-learning textbooks still end with deployment or evaluation, as though future maintenance considerations will somehow take care of themselves.

A Recommendation From the Field

When talking to e-learning managers, it was clear that each had a unique environment, and comparing experiences between companies is a complicated proposition. Out of this complex picture, however, one simple bit of advice emerged as a best practice, recommended again and again during the course of this study. *Track courseware maintenance labor independently of your development labors for new courses.* If you are not doing this already, start today. Otherwise, you won't stand a chance of sorting it out later, and there *will* come a day when you need to.

MITIGATION

Old, New, and Emerging Technologies

The study in this paper had two points of genesis. One was simply the observation that while, in theory, requirements are supposed to be cemented in place prior to development, we in the e-learning community were seeing many maintenance requests that looked like requirement changes in disguise.

The other point of origin came from a paper on handling content that is expected to require frequent updates (Duc, & Haddawy 2004). The authors of this paper outline technical ways to proactively mitigate cost when you know in advance that given areas of content will change frequently. On the high end of the effort scale, ISDs can create data modules (SCORM SCOs) that can be "unplugged" and replaced without unloading the course from the LMS. On the lower end of the effort scale, ISDs can design multimedia presentations that use lower-cost production methods for the portions of content most likely to change.

This prompts the question: if there are design strategies that can mitigate the cost of *content* changes, wouldn't it also be possible to design courseware in ways that reduce the cost of *requirement* changes? Even without knowing the specifics, might it be possible to anticipate the need for change, and design courseware that is inherently flexible, adaptable, and, in general, more change tolerant?

The answer is a qualified "Yes." This paper will close with a brief overview of some technologies that incorporate maintenance management as part of their design concept. A couple of these originated in the technical documents community and may not be familiar to some readers. Others, like SCORM, will be familiar to most. Please note that this section is not intended to recommend any particular approach to reducing costs for maintenance. The particulars of your situation will govern whether any of these (or other approaches) might be cost-effective.

SCORM

We might be accustomed to thinking of SCORM primarily as a reusability standard, since so much of its development was, and continues to be, driven by the reusability vision. But maintainability has also been a key design goal of this standard. It could even be argued that, while some of the SCORM "ilities" are still pending full realization, the maintainability, portability, and interoperability goals of SCORM have, for the most part, been successfully realized.

To understand the relationship of SCORM to maintainability, we have to put it in the context of the e-learning circumstances at the time.

At the time that SCORM was designed, the amount of labor involved in integrating e-learning from multiple vendors into a single point of access was often staggering. This was because web-based e-learning in the 1990s was largely developed under a web-application model, using the same development approach and application architecture as storefront sites, banking sites, and other interactive web sites. Each of these programming/deployment environments had a unique infrastructure of server applications, databases, scripting languages, authentication services, and communication protocols, making each e-learning "application" dependent on a highly specific constellation of resources, incompatible with most other deployment environments.

Without a unifying framework, courseware would never have the flexibility to port well from environment to environment. Every web-based e-learning course was painting itself into a technological corner, married to delivery technology that would eventually become obsolete, leaving the course owner in the position of either funding a costly migration (which often amounted to re-engineering the entire course), supporting defunct technology indefinitely, or discontinuing the product.

One of the key insights of SCORM was that the interactions between courseware and its server could be encapsulated in a single model for communication and navigation. That one key enabler now allows e-learning from any source to migrate/integrate into a single LMS. The cost of a maintenance activity that was once very labor-intensive has been significantly reduced through smart design.

Template-Based Authoring and XML Storage

Template-based authoring has a long history in the e-learning community, earning praise and opprobrium in perhaps equal amounts. The basic idea is to separate content from presentation, meaning that text, images, Flash components, and multimedia (the content) are managed independently of layout, branding, graphic design, and navigation (the presentation). Proponents of template-based authoring point to the streamlined authoring process, where developers focus entirely on content rather than formatting and layout. Opponents point to the constraining effect of working with a library of stock layouts and design concepts.

In recent years, template-based authoring has gotten new life with the emergence of content management systems (CMSs) and workflow tools that can automate business processes. CMSs facilitate automated configuration management and content review, and can provide metrics, traceability, and transparency. With the industry acceptance of XML as a common data storage format, earlier concerns about proprietary formats and interoperability are reduced. Recent efforts in the S1000D and DITA communities to support open standards for training content will make this technology even more attractive and viable, as training content and documentation may soon be able to share common tools, IT infrastructure, metrics, and workflow management.

From a maintenance standpoint, template-based tools facilitate and simplify a wide range of maintenance tasks, the most obvious being changing navigation controls, changing display resolutions, and rebranding (recall that rebranding was the single most frequent maintenance activity among the requirements surveyed). Content transformations can be automated, translation into foreign languages can be simplified, and concepts like controlled vocabulary lists can be used to ensure consistency across training courses, and even between training and technical documentation.

Simplified Technical English

Simplified Technical English (STE) is a standard that is familiar mainly inside the international aerospace community. However, as the SCORM and S1000D communities build compatibility between those standards (Gafford 2007), and S1000D vendors begin to support SCORM, STE may become a standard that we begin to encounter more directly. Simplified Technical English is aimed primarily at improving the usability of text by ensuring that language is used succinctly and consistently. The payoff for courseware maintenance comes largely in translation, where STE can reduce the level of effort by up to 30%, because the consistent use of terms facilitates the use of "translation memories" that cue the translator with previous translations of words or phrases. (Tedopres 2008).

Until this technology becomes more mainstreamed in the e-learning community, STE may remain out of reach for most courseware developers. However, in some environments, STE tools may already be owned by companies that support global products, and these tools might be useful resources in building courseware with a global reach.

CONCLUSIONS

A realistic view of e-learning maintenance must recognize that pressures from policies, learner

requirements, business needs, or the technical environment often force courseware into upgrades that may not have been envisioned at design time. According to the research presented here, a typical course has a 72% chance of encountering at least one of the surveyed requirement changes in a two-year window.

With this in mind, training managers and instructional designers face some up-front decisions with regard to the maintainability of their products. One area of decision-making is in the area of tools and technical approach. Some of the tools and design approaches outlined in this paper may mitigate the cost of particular upgrades down the road. The ROI for these technologies will depend on case-by-case factors.

More important than any single technology, however, is an end-to-end awareness of the product lifecycle, which needs to inform decisions across the spectrum of instructional design. Creating effective, engaging training should remain the principal focus of the instructional designer, but, as an industry, we also need to focus resources on the challenge of building change-tolerant courseware: modular, portable, flexible, and adaptable. These characteristics are the keys to designing for survivability in an environment of constant change.

ACKNOWLEDGEMENTS

The author of this paper would like to thank all the survey participants who contributed to this study, and particularly those participants who also offered to discuss their survey responses over the phone. Thanks also to SCCI for supporting this project, Debby Kalk for assisting with research, Jason Haag for his help distributing the survey, Bernardo Salinas for assisting with statistics, and Mark Gerasch of General Dynamics Information Technology for his advice and support.

REFERENCES

Aviation Industry CBT Committee, (2008). *AICC Guidelines and Recommendations: Training Development Checklist*. Retrieved June 19, 2008, from http://www.aicc.org/docs/AGRs/agr012v15.pdf

Cennamo, K. & Kalk, D. (2005) *Real World Instructional Design*. Belmont, CA: Thomson Learning

Dick, L. & Carey, W. (1996) *The Systematic Design of Instruction* (4th ed.). New York: Harper Collins.

Duc, P. T., & Haddawy, P. (2004) "A Modular Approach to E-Learning Content Creation and Maintenance" in *Advances in Web-Based Learning: ICWL 2004*. New York: Springer.

Gafford, W. (2007). "Using S1000D to Bridge Collaboration Gaps Between Technical Data and Supporting Training Content." *Proceedings of the 2007 Interservice/Industry Training, Simulation & Education Conference*, Orlando, FL.

Glushko, R. J., & McGrath, T. (2008) *Document Engineering: Analyzing and Designing Documents for Business Informatics and Web Services*. Cambridge, MA: MIT Press.

Greer, M. (1992). *Instructional Design Project Management*. Englewood Cliffs, NJ: Educational Technology Publications.

Kemp, J., Morrison, G., & Ross, S. (1998). *Designing Effective Instruction*. Upper Saddle River, NJ: Prentice-Hall.

Navy Integrated Learning Environment, (2007). *ILE Content Developer's Handbook (MPT&ECIOSWIT-ILE-HDBK-1B).* Retrieved June 19, 2008, from https://ile-help.nko.navy.mil/ile/content/policy/handbook.aspx

Revelos, J. (2006). *Justifying E-learning: ROI and Impactful Training*. Retrieved June 19, 2008, from http://www.tatainteractive.com/pdf/ROI.pdf

Tedopres, Inc. (2008), Whitepaper: "HyperSTE, Quality Assurance for Clear Content." Retrieved June 27, 2008, via email to info@tedopres.com

Wright, C. R. (2007). *Adapting Learning Materials for Distance Learning*. Retrieved June 19, 2008, from http://www.col.org/colweb/webdav/site/myjahiasite/shared/docs/KS2007_Adapting-Learning-Materials.pdf